

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

з лабораторної роботи №1

з дисципліни «Вступ до баз даних та інформаційних систем»

на тему:

Створення реляційної бази даних

Виконав(-ла):

студент(ка) групи КМ-02

Савченко С. В.

Керівник:

ст. викладач Бай Ю. П.

ЗМІСТ

ЗАВДАННЯ	2
Основні теоретичні відомості про реляційні бази даних	3
Завдання 1	5
Завдання 2	6
Завдання 3	8

ЗАВДАННЯ

Завдання 1. Спроектувати базу даних, що дозволить відобразити наступні події (5 балів):

Людина танцює під музику.

1a) Визначити сутності та їх атрибути, встановити зв'язки між сутностями. Побудувати ER-діаграму.

1b) Побудувати логічну схему таблиць, використовуючи «crow's foot notation».

1c) За допомогою команд мови SQL створити таблиці в СУБД PostgreSQL. Визначити поля та типи. Первинні та зовнішні ключі створювати окремо від таблиць, використовуючи команду ALTER TABLE.

Завдання 2. Згенерувати базу даних з книги Б. Форта та виконати запити (потрібні для виконання завдань файли *create.txt*, *populate.txt* можна завантажити, наприклад, з <https://github.com/alinxSorcere/SQL-in-10-minutes-with-notes.git>) (6 балів):

2a) Скільки продано найдорожчого товару?

2b) Вивести імена покупців, що мають e-mail та живуть в USA.

2c) Вивести PROD_ID товарів та імена постачальників для тих товарів, що були продані. Результат вивести у верхньому регістрі, як єдине поле products_sold.

Завдання 3. Виконати запити 2a), 2b) з попереднього завдання, використовуючи операції реляційної алгебри Кодда та агрегатні функції мови SQL (4 бали).

Основні теоретичні відомості про реляційні бази даних

1970 р., Едгар Кодд, математик-програміст IBM.
Codd E.F., A Relational Model of Data for Large Shared Data Banks.

- застосував термін "**реляційна модель даних**";
- запропонував використовувати для обробки даних апарат **теорії множин**: вибірка, проєкція,
- об'єднання, перетин, різниця, декартів добуток;
- показав, що будь-яке представлення даних
- зводиться до сукупності двовимірних таблиць особливого виду, відомого в математиці як відношення (**relation**).

select * from Products ;				
PROD_ID	VEND_ID	PROD_NAME		PROD_PRICE
BR01	BRS01	8 inch teddy bear	...	5,99
BR02	BRS01	12 inch teddy bear	...	8,99
BR03	BRS01	18 inch teddy bear	...	11,99
BNBG01	DLL01	Fish bean bag toy	...	3,49
BNBG02	DLL01	Bird bean bag toy	...	3,49
BNBG03	DLL01	Rabbit bean bag toy	...	3,49
RGAN01	DLL01	Raggedy Ann	...	4,99
RYL01	FNG01	King doll	...	9,49
RYL02	FNG01	Queen doll	...	9,49

Відношення (relation) — фундаментальне поняття реляційної моделі даних.

Реляційна база даних – це база даних, у якій дані представлені у вигляді таблиць (відношень).

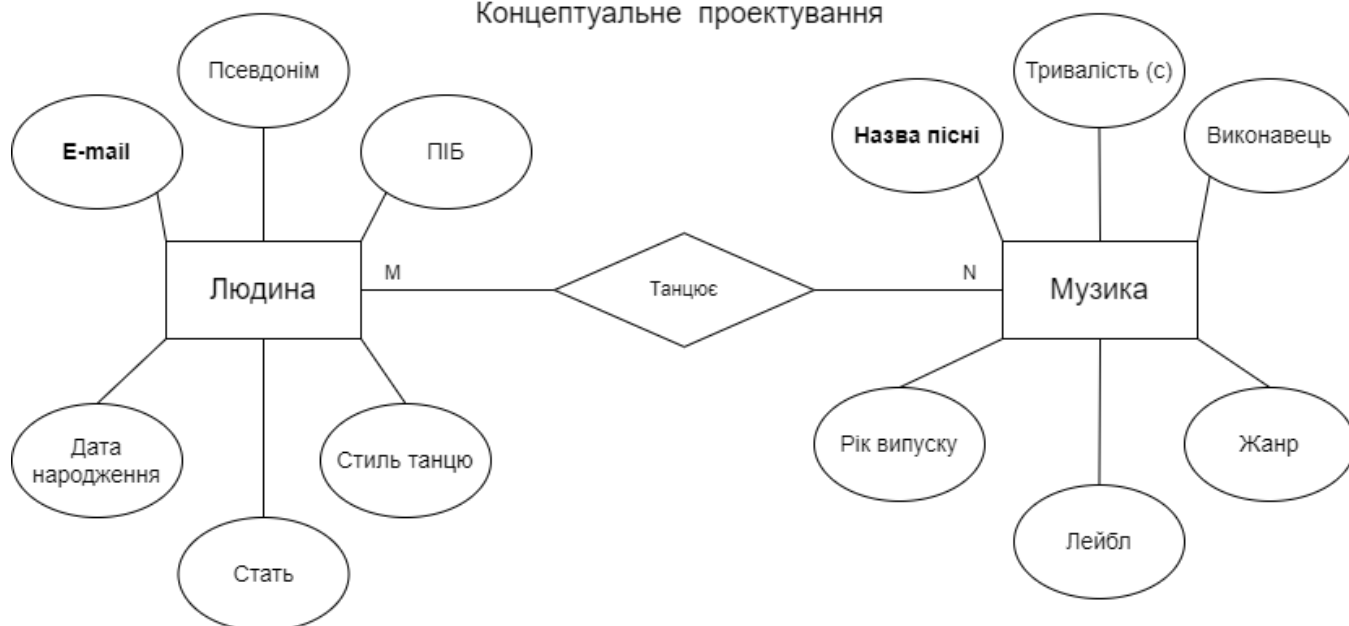
N -арним відношенням R , або відношенням R степеня n , називають підмножину декартового добутку множин D_1, D_2, \dots, D_n ($n \geq 1$), які не обов'язково є різними. Вхідні множини D_1, D_2, \dots, D_n називають **доменами (sets)**.

Завдання 1

Людина танцює під музику

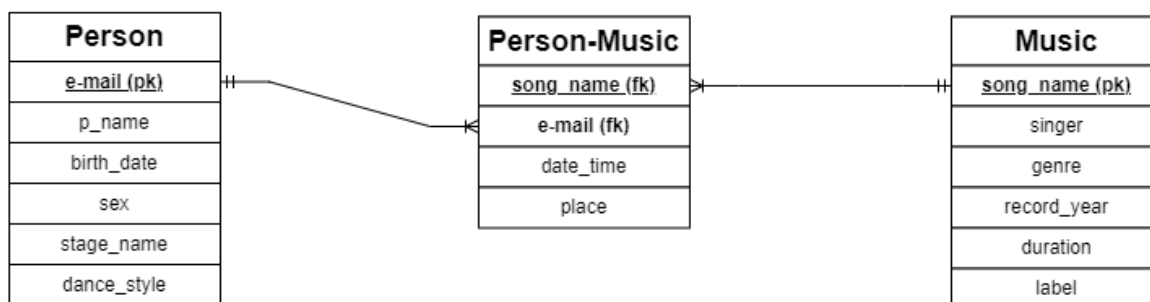
1 ЕТАП

Концептуальне проектування



2 ЕТАП

Логічне проектування



3 ЕТАП

Фізичне проектування

```

CREATE TABLE Person (
  e-mail char(20) NOT NULL,
  p_name char(20) NOT NULL,
  birth_date date NULL,
  sex char(1) NOT NULL,
  stage_name char(40) NOT NULL,
  dance_style char(10) NOT NULL
);
  
```

```

CREATE TABLE Person-Music (
  song_name char(40) NOT NULL,
  e-mail char(20) NOT NULL,
  date_time date NULL,
  place char(25) NOT NULL
);
  
```

```

CREATE TABLE Music (
  song_name char(40) NOT NULL,
  singer char(20) NOT NULL,
  genre char(10) NOT NULL,
  record_year date NULL,
  duration int,
  label char(20) NOT NULL
);
  
```

```
ALTER TABLE Person ADD PRIMARY KEY (e-mail);
```

```
ALTER TABLE Music ADD PRIMARY KEY (song_name);
```

```
ALTER TABLE Person CONSTRAINT FK_Person_Music FOREIGN KEY
(song_name) REFERENCES Music (song_name);
```

Завдання 2

Базу даних було встановлено на перших лабораторних заняттях.

а) Скільки продано найдорожчого товару?

```

2 SELECT SUM(quantity) FROM orderitems
3 WHERE item_price = (SELECT MAX(item_price) FROM orderitems)

```

Data output		Messages	Notifications
	sum bigint		
1	15		

Рис.1 – результат виконання запиту №1

Відповідь: Було продано 15 одиниць найдорожчого товару.

б) Вивести імена покупців, що мають e-mail та живуть в USA.

```

5 --2b
6 SELECT cust_contact FROM customers
7 WHERE (cust_email IS NOT NULL) AND (cust_country = 'USA')
8

```

Data output		Messages	Notifications
	cust_contact character (50)		
1	John Smith		
2	Jim Jones		
3	Denise L. Stephens		

Рис.2 – результат виконання запиту №2

- с) Вивести PROD_ID товарів та імена постачальників для тих товарів, що були продані. Результат вивести у верхньому регістрі, як єдине поле products_sold.

```

10 SELECT UPPER(products.prod_id || ' - ' || vendors.vend_name) AS products_sold
11 FROM products JOIN vendors
12 ON products.vend_id = vendors.vend_id
13 WHERE prod_id IN (SELECT prod_id FROM orderitems)

```

Data output Messages Notifications

	products_sold text	
1	BR02 - BEARS R US	
2	BR01 - BEARS R US	
3	BR03 - BEARS R US	
4	RGAN01 - DOLL HOUSE IN...	
5	BNBG02 - DOLL HOUSE IN...	
6	BNBG03 - DOLL HOUSE IN...	
7	BNBG01 - DOLL HOUSE IN...	

Рис. 3 – результат виконання запиту №3

Завдання 3

Виконати запити 2a), 2b) з попереднього завдання, використовуючи операції реляційної алгебри Кодда та агрегатні функції мови SQL.

2a)

$$R1 \leftarrow \text{MAX}(\pi_{\text{itemprice}}(\text{orderitems}))$$

$$R2 \leftarrow \sigma_{\text{itemprice}=R1}(\text{orderitems})$$

$$R3 \leftarrow \text{SUM}(\pi_{\text{quantity}}(R2))$$

2	SELECT SUM(quantity) FROM orderitems
3	WHERE item_price = (SELECT MAX(item_price) FROM orderitems)

Data output	Messages	Notifications
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> </div>		
	sum	🔒
	bigint	
1		15






2b)

$$R1 \leftarrow \sigma_{\text{cust_email} \neq \text{NULL} \wedge \text{cust_country}='USA'}(\text{customers})$$

$$R2 \leftarrow \pi_{\text{cust_contact}}(R)$$

```
5  --2b
6  SELECT cust_contact FROM customers
7  WHERE (cust_email IS NOT NULL) AND (cust_country = 'USA')
8
```

Data output Messages Notifications

		       	
		cust_contact	
		character (50)	
1	John Smith		
2	Jim Jones		
3	Denise L. Stephens		